

**Name: Ali Bilal**  
**Roll No: P20-0077**  
**Section: 6C**

## LAB TASK 6

```
In [188]: import pandas as pd
          from sklearn.preprocessing import LabelEncoder
          from sklearn.ensemble import RandomForestClassifier
          from sklearn.model_selection import train_test_split
          from sklearn.neural_network import MLPClassifier
          from sklearn.metrics import recall_score, f1_score
```

```
In [164]: # Replace the missing value with the mode
          mode_embarked = df['embarked'].mode()[0]
          print(mode_embarked)
          df['embarked'].fillna(mode_embarked, inplace=True)
          # df
          S
```

```
In [165]: missing_values_count = df['embarked'].isnull().sum()
          print(missing_values_count)
          0
```

```
In [167]: df
```

```
Out[167]:
```

	passenger_id	pclass	name	sex	age	sibsp	parch	ticket	fare	embarked	boat	body	home.dest	survived
0	1216	3	Smyth, Miss. Julia	0	29.519847	0	0	335432	7.733300	Q	13	NaN	NaN	1
1	699	3	Cacic, Mr. Luka	1	38.000000	0	0	315089	8.662500	S	NaN	NaN	Croatia	0
2	1267	3	Van Impe, Mrs. Jean Baptiste (Rosalie Paula Go...	0	30.000000	1	1	345773	24.150000	S	NaN	NaN	NaN	0
3	449	2	Hocking, Mrs. Elizabeth (Eliza	0	54.000000	1	3	29105	23.000000	S	4	NaN	Cornwall / Akron, Ohio	1

```
In [150]: df = pd.read_csv('titanic_train.csv')
```

```
In [151]: df
```

```
Out[151]:
```

	passenger_id	pclass	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	boat	body	home.dest	survived
0	1216	3	Smyth, Miss. Julia	female	NaN	0	0	335432	7.7333	NaN	Q	13	NaN	NaN	1
1	699	3	Cacic, Mr. Luka	male	38.0000	0	0	315089	8.6625	NaN	S	NaN	NaN	Croatia	0
2	1267	3	Van Impe, Mrs. Jean Baptiste (Rosalie Paula Go...	female	30.0000	1	1	345773	24.1500	NaN	S	NaN	NaN	NaN	0
3	449	2	Hocking, Mrs. Elizabeth (Eliza Needs)	female	54.0000	1	3	29105	23.0000	NaN	S	4	NaN	Cornwall / Akron, OH	1
4	576	2	Veal, Mr. James	male	40.0000	0	0	28221	13.0000	NaN	S	NaN	NaN	Barre, Co Washington, VT	0
5	1083	3	Olsen, Mr. Henry Margido	male	28.0000	0	0	C 4001	22.5250	NaN	S	NaN	173.0	NaN	0

```
Activities Brave Web Browser 15:12 20 ماح 15
localhost:8888/notebooks/Documents/AI%20Lab/Lab%206/Untitled.ipynb... Python 3 (ipykernel)

In [166]: mean_value = df['age'].mean()
          print(mean_value)
          29.519847189349115

In [167]: df['age'].fillna(mean_value, inplace=True)
          missing_values_count = df['age'].isnull().sum()
          print(missing_values_count)
          0

In [142]: le = LabelEncoder()
          # apply Label Encoding to the 'sex' column
          sex_encoded = le.fit_transform(df['sex'])

          # create a new dataframe with the encoded 'sex' column
          df_encoded = pd.DataFrame({'sex_encoded': sex_encoded})

          # print the resulting dataframe
          print(df_encoded)

          -----
          TypeError                                Traceback (most recent call last)
          /tmp/ipykernel_6205/3383266719.py in <module>
              2
              3 # apply Label Encoding to the 'sex' column
          ----> 4 sex_encoded = le.fit_transform(df['sex'])
              5
              6 # create a new dataframe with the encoded 'sex' column

          TypeError: 'NoneType' object is not subscriptable
```



```
In [152]: # df = df['age']
missing_values_count = df['age'].isnull().sum()
print(missing_values_count)
```

174

```
In [153]: mean_value = df['age'].mean()
print(mean_value)
```

29.519847189349115

```
In [154]: df['age'].fillna(mean_value, inplace=True)
missing_values_count = df['age'].isnull().sum()
print(missing_values_count)
```

1

```
In [155]: le = LabelEncoder()

# apply Label Encoding to the 'color' column
df['sex'] = le.fit_transform(df['sex'])

# print the resulting dataframe
print(df)
```

```
In [157]: pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)

# view the 'Name' column
print(df['cabin'])
```

0	NaN
1	NaN
2	NaN
3	NaN
4	NaN
5	NaN
6	NaN
7	NaN
8	NaN
9	NaN
10	C82
11	D15
12	C50
13	NaN
14	NaN
15	NaN
16	NaN
17	NaN
18	E33
19	NaN

```
In [171]: df = df.drop('name', axis=1)
```

```
In [172]: df = df.drop('boat', axis=1)
```

```
In [173]: df
```

```
Out[173]:
```

	passenger_id	pclass	sex	age	sibsp	parch	ticket	fare	embarked	survived
0	1216	3	0	29.519847	0	0	335432	7.733300	Q	1
1	699	3	1	38.000000	0	0	315089	8.662500	S	0
2	1267	3	0	30.000000	1	1	345773	24.150000	S	0
3	449	2	0	54.000000	1	3	29105	23.000000	S	1
4	576	2	1	40.000000	0	0	28221	13.000000	S	0
5	1083	3	1	28.000000	0	0	C 4001	22.525000	S	0
6	898	3	1	19.000000	0	0	LINE	0.000000	S	0
7	560	2	0	30.000000	0	0	250648	13.000000	S	1
8	1079	3	0	22.000000	0	0	347085	7.775000	S	1
9	908	3	0	21.000000	1	0	4137	9.825000	S	0
10	313	1	1	27.000000	0	2	113503	211.500000	C	0

```
In [174]: missing_values_count = df['embarked'].isnull().sum()
print(missing_values_count)
0
```

```
In [171]: df = df.drop('name', axis=1)
```

```
In [172]: df = df.drop('boat', axis=1)
```

```
In [173]: df
```

```
Out[173]:
```

	passenger_id	pclass	sex	age	sibsp	parch	ticket	fare	embarked	survived
0	1216	3	0	29.519847	0	0	335432	7.733300	Q	1

```
In [176]: df = df.drop('ticket', axis=1)
```

```
In [177]: X = df.drop("survived", axis=1)
y = df["survived"]

# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Check the shape of the training and test sets
print("Training set shape:", X_train.shape)
print("Test set shape:", X_test.shape)

Training set shape: (680, 8)
Test set shape: (170, 8)
```

```
In [178]: missing_values_count = df['survived'].isnull().sum()
print(missing_values_count)
```

```
0
```

